

MODELING AND SIMULATION FOR RF SYSTEM DESIGN

# Modeling and Simulation for RF System Design

by

**RONNY FREVERT**

*Fraunhofer Institute for Integrated Circuits, Dresden, Germany*

**JOACHIM HAASE**

*Fraunhofer Institute for Integrated Circuits, Dresden, Germany*

**ROLAND JANCKE**

*Fraunhofer Institute for Integrated Circuits, Dresden, Germany*

**UWE KNÖCHEL**

*Fraunhofer Institute for Integrated Circuits, Dresden, Germany*

**PETER SCHWARZ**

*Fraunhofer Institute for Integrated Circuits, Dresden, Germany*

**RALF KAKEROW**

*Nokia Research Center, Bochum, Germany*

and

**MOHSEN DARIANIAN**

*Nokia Research Center, Bochum, Germany*

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 10 0-387-27584-3 (HB)  
ISBN 13 978-0-387-27584-0 (HB)  
ISBN 10 0-387-27585-1 (e-book)  
ISBN 13 978-0-387-27585-7 (e-book)

---

Published by Springer,  
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

*www.springeronline.com*

The book and the included CD-ROM contain models which may be used for simulation purposes. The user accepts full responsibility for the use of these models.

The names of software products used in this book are trademarks of their respective producers.

*Printed on acid-free paper*

All Rights Reserved

© 2005 Springer

No part of this work may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, microfilming, recording or otherwise, without written permission from the Publisher, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work.

Printed in the Netherlands.

# Contents

Preface	ix
Acknowledgments	xi
1. INTRODUCTION	1
2. DESIGN FLOW OVERVIEW	7
2.1 Design Levels	7
2.2 Top-down System Design	9
2.3 Bottom-up Verification	11
3. SIMULATION TOOLS IN SYSTEM DESIGN	15
3.1 Use of Simulation Tools within the Design Flow	15
3.2 Specific Simulation Algorithms of RF Simulators	17
3.3 Criteria of the Simulator Selection	21
3.4 Internet Resources for Simulation Tools	23
4. SYSTEM LEVEL MODELING	25
4.1 System Level Simulation	25
4.2 Simulation Technology of System Level Simulators	26
4.3 Complex Baseband Simulation	27
4.3.1 Principle	27
4.3.2 Example for baseband simulation	30
4.3.3 Restrictions and advantages of baseband modeling	30
4.4 Model Libraries for System Simulation	31
4.5 Creation of Own Primitive and Hierarchical Models	33

4.5.1	SPW modeling example	33
5.	VHDL-AMS FOR BLOCK LEVEL SIMULATION	39
5.1	Introduction	39
5.2	VHDL-AMS Standardization	40
5.3	A Simple Block Level Example – Analog PLL	41
5.3.1	Mathematical models of basic blocks	42
5.3.2	Structural description of the PLL circuit in VHDL-AMS	44
5.3.3	VHDL-AMS description of basic blocks	47
5.4	Summary	50
6.	INTRODUCTION TO VHDL-AMS	51
6.1	Aim of this Introduction	51
6.2	Repetition of Basics of VHDL 1076-1993	52
6.2.1	Design units	52
6.2.2	Logical libraries and compilation of design units	56
6.2.3	Concurrent statements	60
6.2.4	A simple pure digital example – divider	65
6.3	Conservative Systems Description	66
6.3.1	Network analysis problem	67
6.3.2	Nature, terminal and branch quantity declarations	71
6.3.3	Simultaneous statements and free quantity declarations	78
6.3.4	Example of a conservative system – A-law companding	85
6.3.5	Attributes in VHDL-AMS	88
6.3.6	Example – higher order lowpass filter	103
6.4	Description of Nonconservative Systems	105
6.5	Mixed-Signal Simulation	107
6.5.1	Attributes for mixed-signal modeling	108
6.5.2	Mixed-signal simulation cycle	114
6.6	Analysis Domains	116
6.6.1	Supported domains	116
6.6.2	Small-signal and noise domain simulation	118
6.7	Summary	124
7.	SELECTED RF BLOCKS IN VHDL-AMS	127
7.1	Library Overview	127
7.2	Signal Sources	128
7.2.1	Independent sources	128
7.2.2	Modulated sources	130
7.2.3	Wobble generator	133
7.2.4	Pseudorandom binary source	135
7.3	Basic RF Building Blocks	137
7.3.1	Low-noise amplifier	137

7.3.2	Mixer	142
7.3.3	Charge pump	146
7.3.4	Analog VCO	150
7.3.5	Digital VCO	153
7.3.6	Filters	157
7.3.7	Switch	163
7.3.8	General n-bit A/D and D/A converter	164
7.3.9	Simple channel	169
7.4	Measurement and Observation Units	174
7.4.1	Peak detector	174
7.4.2	Frequency measurement unit	175
7.4.3	Power meter	178
7.5	Block Level Example of a Linear PLL	183
8.	MACROMODELING IN VHDL-AMS	191
8.1	Introduction	191
8.2	General Methodology	191
8.3	Input and Output Stages	194
8.3.1	Input stages	194
8.3.2	Output stages	197
8.4	OpAmp Macromodel	199
9.	COMPLEX EXAMPLE: WLAN RECEIVER	203
9.1	Introduction	203
9.2	Example Specification	204
9.3	Example Modeling	207
9.4	Example Calibration	211
9.5	Example Verification	214
10.	MODELING OF ANALOG BLOCKS IN VERILOG-A	219
10.1	Introduction	219
10.2	Writing Custom Behavioral Models	220
10.2.1	Verilog-A principles	220
10.2.2	LNA modeling example	222
10.2.3	Creating a Verilog-A model	226
10.3	Overview of the Cadence Model Library rfLib	231
10.4	Modeling and Simulation of a WLAN Receiver	236
10.4.1	WLAN receiver modeling using Cadence libraries	237
10.4.2	Simulation of the WLAN receiver	240
11.	CHARACTERIZATION FOR BOTTOM-UP VERIFICATION	247
11.1	Concept of Characterization	247
11.2	RF Characteristics and Parameters	248

11.3	Application of Characterization	252
11.4	Example Characterization of an LNA	254
11.5	Characterization Environment	258
11.6	Characterization Using the OCEAN Script Language	262
11.6.1	Creation of the testbench schematic	262
11.6.2	Analysis settings and simulation	263
11.6.3	Combination and extension of the OCEAN scripts	266
12.	ADVANCED METHODS FOR OVERALL SYSTEM SPECIFICATION AND VALIDATION	271
12.1	Gap between System Level and Block Level Simulation	271
12.2	File Coupling of Simulators	272
12.3	Direct Cosimulation of System Level and Analog Simulators	273
12.4	Generated Black Box Models	279
	References	285
	Index	287

## Preface

Many books have been published in recent years that focus on wireless communication systems, with some focused on modeling and simulation. This book is aimed at the special topic of modeling for RF system design. Very high carrier frequencies together with long observation periods result in extremely large computation times and requires, therefore, specialized modeling methods and simulation tools on all design levels from system down to circuit level. To illustrate the application of these methods and usage of the tools the book includes numerous models and extensive examples. Therefore the book is addressed to graduate students and industrial professionals who are engaged in communication system design and want to gain insight into the system structure by own simulation experiences.

The tools and languages for hardware description of VLSI circuits have changed over the years. Nevertheless models are provided on a CD-ROM included with this book because models are necessary to reproduce, understand and explore the real world behavior on a simulation platform. VHDL-AMS and Verilog-A are chosen as description languages which are an IEEE standard and a quasi industrial standard respectively. In spite of deviations within language implementations in different simulation tools, the provided mathematical background to each individual model should enable a large audience of readers to use these models. Moreover the given introduction into the syntactic elements of the language VHDL-AMS allows to modify the given examples to special needs.

*The authors*

## **Acknowledgments**

This book is the result of many years of fruitful project cooperation between Nokia Research Center, the Fraunhofer Institute for Integrated Circuits and other partners. After common discussions and successful research in the field of modeling methodology for wireless system design we were convinced that it is time to publish our approaches, methods and results together with illustrating examples.

The authors are grateful to all colleagues inside and outside of our organizations for sharing their knowledge during discussions and to all supporters who helped with their valuable hints and corrections to complete the work on this book. Especially we wish to thank Dean Hobson from Mentor Graphics who carefully read the manuscript and was always prepared to discuss matters of language and content. Also, we would like to thank Mark de Jongh for his encouraging hints and the management task to publish this book. This also includes of course the staff at Kluwer publishers who produced this book in a very professional way.

# Chapter 1

## INTRODUCTION

Modern telecommunication systems are highly complex from an algorithmic point of view. The complexity continues to increase due to advanced modulation schemes, multiple protocols and standards, as well as additional functionality such as color displays, personal organizers, navigation aids, cameras, and audio-visual support.

At the same time both silicon area – which means costs – and power consumption of the devices have to be reduced and the design time shortened. This is inevitable to keep profitability in this fast evolving high volume consumer market.

These conflictive demands force the need for efficient design and verification methods. To have short and reliable design cycles, verification is necessary very early in the design process. Modeling and simulation need to accompany the design steps from the specification to the overall system verification in order to bridge the gaps between system specification, system simulation, and circuit level simulation. Therefore this book contains application-oriented training material for RF designers which combines the presentation of a mixed-signal design flow, an introduction into the standardized powerful hardware description language VHDL-AMS, and the application of commercially available simulators. The focus lies on RF specific modeling and simulation methods and the consideration of system and circuit level descriptions.

An early version of some parts of this book, especially some of the VHDL-AMS models, has been tested in a Nokia-internal course with about 50 designers. In this course a web-based education and simulation environment has been used, developed in a European research project LIMA (Learning Platform in Microelectronics Applications).

The challenges for the designer are especially demanding in the face of mixed-signal (analog/digital) and multi-domain (RF/baseband) systems. Today's wireless communication systems use sophisticated modulation and coding techniques to transmit the information at very high carrier frequencies. Modulation and coding is typically realized in the Digital Signal Processing (DSP) subsystem, which is also called baseband signal processing. The RF front-end provides the interface between baseband (some MHz) and the RF transmission channel (some GHz).

The DSP part uses more than 95% of the total amount of transistors. System level simulators are used for the verification of the DSP algorithms. Efficient simulation algorithms are applied to simulate the complete transmit path from the transmitter to the receiver. DSP designers often assume that the analog part is an ideal device. On the other hand RF designers perform analog simulations to design and verify the RF subsystem without information regarding the DSP part. This is why the common evaluation of the RF and the DSP part becomes increasingly important. This ensures that the RF part fulfills the system requirements without over-dimension, which means the interaction between both parts is respected without the need to include a safety margin in the specification of the RF part.

RF circuits and systems possess special characteristics that need to be considered in modeling and simulation, which are

- very high carrier frequency on the one hand and comparatively low signal bandwidth on the other,
- presence of weak nonlinearities,
- importance of noise considerations and the signal-to-noise ratio (SNR),
- necessity to simulate a large number of sample points or data bits in order to compute distortion measures, for example bit error rates (BER).

For RF systems to handle these characteristics specially suited modeling methods and simulation algorithms have been developed. They will be introduced during the course of this book and demonstrated with examples.

A number of simulation tools are on the market that specialize in RF circuits. Since we want to widen the scope on a design flow from system to circuit level with attention to mixed-signal aspects, we used a collection of different commercially available simulation tools in the book

- ADVance MS of Mentor Graphics
- SpectreRF of Cadence
- SPW of CoWare
- MATLAB of The MathWorks

Many other tools currently available on the market could have been used, but the modeling methods and simulation principles remain the same. An introduction into the usage of the tools goes beyond the scope of this book. For support on the tools, refer to the help function or the online help of the tool providers. It is also not intended to include schematic entry and layout tools.

Modeling of RF systems ranges from system-level signal-flow oriented models (for example MATLAB/Simulink) over mixed-signal block oriented models (for example VHDL-AMS) to circuit-level descriptions (for example SpectreRF). Therefore a modeling flow, covering different levels of abstraction, as well as modeling languages and libraries are essential topics of the book (Figure 1-1). A special focus lies on the mixed-signal simulator-independent modeling language VHDL-AMS.

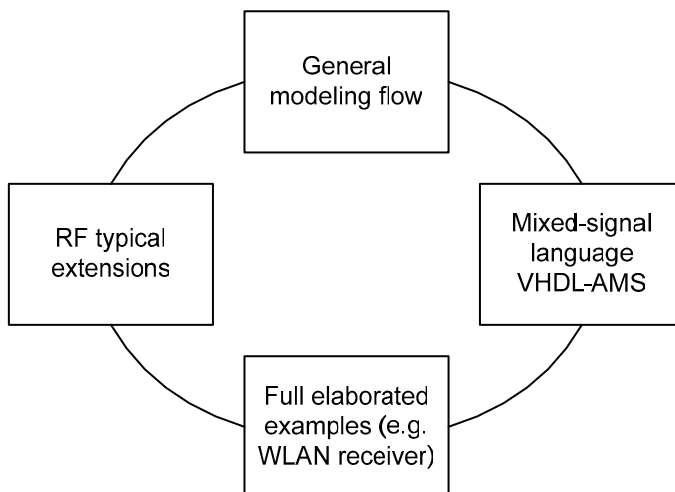


Figure 1-1. Overview of the main topics of the book

Modeling and simulation methods need to be oriented on existing design flows in order to establish them in industrial use. Hence we propose a modeling and simulation flow that follows the V-diagram as a commonly accepted design paradigm (see Chapter 2). The material in this book is structured accordingly. Chapter 2 provides an overview of different levels of abstraction, the top-down and bottom-up methodologies. Specific simulation algorithms and various simulation tools for different phases of RF system design are introduced in Chapter 3.

The first direction of the design flow is top-down. That means we start with specifications at the system level. Chapter 4 describes how RF components can be modeled in system level simulators such as CoCentric, SPW or MATLAB. It is focused on the development of RF-specific system models.

After initial architectural decisions, specifications for the subsystems are derived and an abstract (less detailed) behavioral model of the RF subsystem can be developed for simulation. This model is improved and becomes more detailed during the design process. On this architecture or block level, mixed-signal simulations are often necessary because the partition into analog and digital parts is not yet clear and different architectures have to be explored. At this point in the book we introduce VHDL-AMS as an important language that supports digital, analog, and mixed-signal modeling and simulation. It is a strict superset of the digital VHDL 1076-1993. Chapter 6 is aimed at designers with knowledge of standard digital VHDL 1076-1993. The reader should be able to understand and use the provided models, change and refine them, as well as develop own simple models.

A library of RF block level models in VHDL-AMS is fully documented in Chapter 7. The enclosed CD-ROM contains the complete source code of this model library. Important basic RF building blocks are included subdivided into source, processing and measurement blocks. Chapter 8 introduces the macromodeling principle with examples in VHDL-AMS.

In Chapter 9 the complex design example of a WLAN receiver according to the standard IEEE 802.11a is assembled from basic building blocks of the previous chapters. Using the modeling flow methodology from the previous chapters the example is modeled in VHDL-AMS, optimized using circuit level simulation, and verified by system level simulation. Thereby it is shown, how the realistic design task of developing a receiver front-end can be supported by modeling and simulation.

The next step in the top-down design flow is the implementation of blocks as circuits. At this level, circuit simulators are available with dedicated support for RF analysis and depiction modes. The custom IC design environment from Cadence and its analog RF simulator SpectreRF are important tools in RF circuit design. SpectreRF uses Verilog-A for behavioral modeling, which is the analog part of Verilog-AMS. A library of Verilog-A models for typical RF building blocks is provided by Cadence. Chapter 10 demonstrates the use of this library for RF system modeling. An example of modeling in Verilog-A is provided.

Bottom-up techniques are used next in the design flow to verify whether design goals are met with the implemented system. The characterization of circuit level descriptions allows the refinement of behavioral models for system level simulation. It is also applied to generate data for the component documentation and reuse. Characterization environments are discussed in the Chapter 11. The characterization environment is used to extract RF specific parameters of circuit designs and to validate the respective behavioral models. An overview of parameters, which can be extracted for RF components, is provided. A characterization example is demonstrated by using SpectreRF and OCEAN scripts.

As a last step in the design flow, system verification is necessary with the back-annotated knowledge of the circuit properties in the refined models. Solutions which will bring analog and system level simulators together are introduced in the last Chapter 12. Black box modeling uses a special kind of characterization to generate nonlinear transfer functions of a complete RF front-end. The transfer functions are stored in files which are read from special black box models in the system level simulator. Another method is co-simulation, which couples analog and system level simulators. The principles of both approaches are explained and illustrated by examples for the Cadence design environment. Advantages and disadvantages of the different approaches are discussed.

To summarize, the training material comprises up-to-date knowledge of modeling and simulation for the RF system design of modern telecommunication systems. The introduction of a general modeling flow is supplemented by RF specific simulation algorithms. Commercially available tools are used to demonstrate how RF system design can be supported and improved by means of modeling and simulation. A second major part is the introduction of VHDL-AMS as a standardized hardware description language with increasing importance. Because it is the mixed-signal extension of the well-established language VHDL it is expected to be used for RF and system design tasks in the near future.

In this application-oriented book the teaching material, which introduces the concepts and theoretical background, is followed by illustrative examples and sources of further information. Many simulation examples are shown with extensive solutions. Thus if the reader has access to the required simulation tools he is able to reproduce the example solution, modify it and thereby gain own experiences with modeling and simulation of RF systems. This book establishes a comprehensive training course in a technologically critical area.

## Chapter 2

# DESIGN FLOW OVERVIEW

### 2.1 Design Levels

Functionality and architecture of electronic devices can be very complex. The systems may consist of analog and digital hardware together with software parts. A telecommunication system contains for example:

- An analog front-end to the physical transmission channel
- Digital hardware for coding and modulation
- General purpose or signal processors for control, user interface and transmission protocol handling

Many designers with specialization in different areas are involved in design and implementation. Several design steps are necessary to realize a system concept on silicon. The design process can be classified in several design levels as shown in Figure 2.1.

Each design level is associated with certain design tasks concerning the whole system or system parts. Starting from system level the design description becomes more and more detailed in a design step. CAD tools support the designer at each level.

The system level is the first design level beginning with an idea of the desired system. This level is also called concept engineering. The system concept and main algorithms are described at a very abstract level without information about the implementation of algorithms. For example, the coding algorithm to be used for data transmission is specified, but it is not decided to implement the coder in hardware or software.

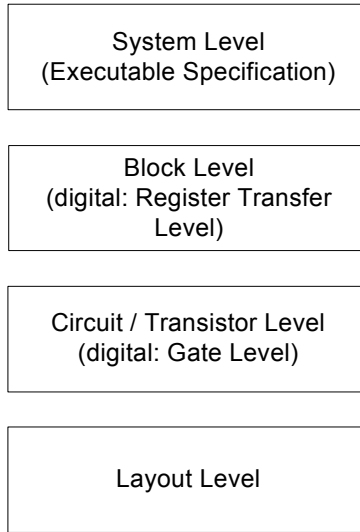


Figure 2-1. Design levels

The system specification can be developed on a sheet of paper. More powerful is an executable specification supported by system-level simulators (for example CoCentric System Studio, MATLAB, and SPW). It allows the evaluation of the selected algorithms and provides a reference model for following design steps.

The system is now partitioned into several hardware (analog or digital) and software subsystems. This design level is named Block Level or Register Transfer Level (RTL) in the digital area. The description of the subsystems at this level contains more detail about the design architecture. At this level the design consists of different blocks, for example multiplier, adder, register, A/D converter, analog filter and amplifier.

Digital and mixed-signal hardware description language (HDL) simulators support the block level design. Commonly used modeling languages in this area are VHDL-AMS and Verilog-AMS. The design of hardware/software systems is further supported by special tools, for example instruction set simulators (ISS).

The third design level is called gate level in the digital domain and circuit level in the analog domain. The blocks of the system are now represented by netlists containing gates or active and passive analog elements. Gate level models can be generated from RTL descriptions by logic synthesis. In the analog design, the circuits are still designed manually.

Gate level or circuit simulation is used to evaluate the design at block level. In the digital domain a timing analysis can be executed, and the blocks

are still described in VHDL and Verilog. Circuit simulators such as SPICE and Spectre are used in the analog domain to analyze the behavior of the designed block.

Based on the gate level or circuit netlist and data of the circuit technology the layout of the circuit is designed. The design is now represented as polygons at different layers of an integrated circuit. In the digital domain this step is well-automated. The tools will check if the design rules for a specified circuit technology are fulfilled. In the analog domain further manual optimization of layout may be necessary, for example to minimize crosstalk between signals or to achieve a symmetric design. Tools that extract parasitic effects that originate from layout also support the layout verification.

## 2.2 Top-down System Design

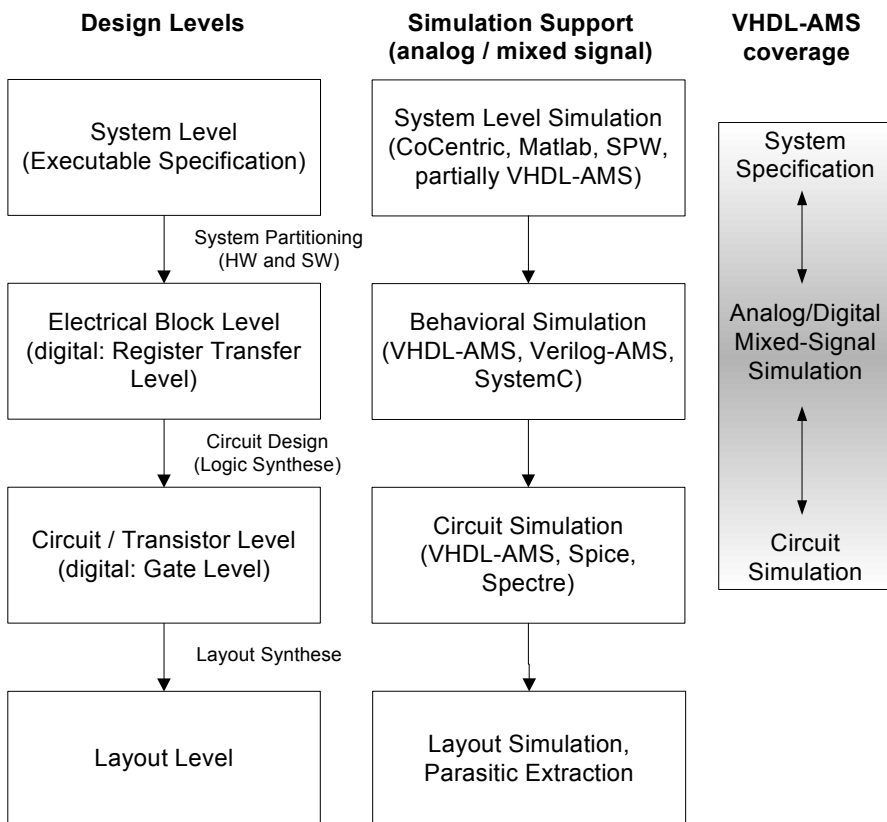


Figure 2-2. Top-down design and simulation support

Top-down design is a method of designing an electronic system that starts with the complete system concept and then breaks it down into smaller and smaller components (see Figure 2-2).

The first design level at which top down design starts is the system level. For telecommunication systems it is here that is specified which algorithms are used to transmit data from the signal source at point A to a sink at point B. Algorithms which are specified at this level may be for example:

- data structure and protocol
- forward error correction techniques (FEC)
- modulation techniques (QPSK, QAM, GMSK, OFDM)
- channel equalization and synchronization

The system level design is supported by system level simulation. Efficient simulation techniques (for example event driven or data stream driven simulation) allow the simulation of the complete transmission system. The simulation also includes a model of the transmission channel (additive white Gaussian noise, AWGN, or mobile channels with fading). The goal of the system design is an overall system specification. If a system level simulation model exists, it can be used as an "executable specification" (see Figure 2-3).

If the system level specification was successfully verified within a system level simulation the system is partitioned. The algorithms of the system can be implemented in different ways:

- analog hardware
- digital hardware
- software

The second design level is named Block Level or in the digital area Register Transfer Level. The system is now partitioned into components and subsystems. Now parameters of the components can be specified.

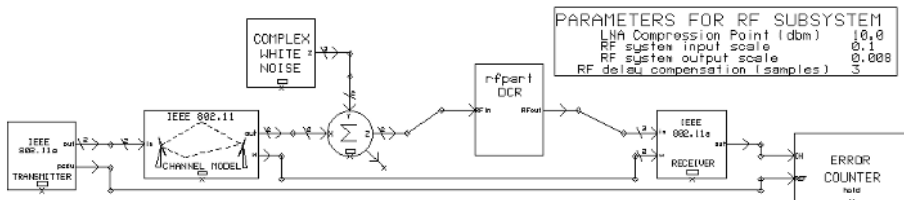


Figure 2-3. Top level schematic of a WLAN system simulation model (SPW)

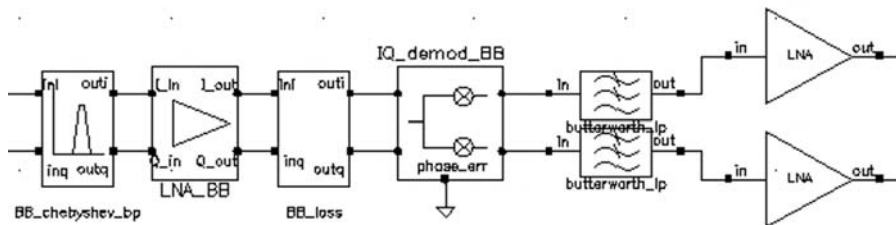


Figure 2-4. Schematic of the RF subsystem (direct conversion receiver)

Figure 2-4 shows for example the block level schematic of the RF subsystem of the WLAN receiver. At system level the RF subsystem was specified either with ideal parameters or with parameters like noise level, gain and linearity. Now it is broken down into its components (filter, amplifier and mixers) which must be parameterized.

At block level we use behavioral models for the simulation of the subsystems. For the analog and mixed-signal area, models can be written in VHDL-AMS and Verilog-AMS. For pure analog simulation, additional languages (for example SpectreHDL) are provided with the simulation tools. The simulation at block level is used to verify whether the block level realization of the subsystem meets the system level requirements.

After the blocks are specified, the circuit design can start. In the digital area, gate level designs can be generated automatically from behavioral models. However for analog blocks there are still no synthesis tools available. So the analog designers must create the transistor level implementation of the components manually. This is supported by transistor level simulation. The block level simulation models can be reused as testbench or reference models if the circuit level simulator supports behavioral modeling languages. Verilog-AMS and VHDL-AMS simulators often support the simulation of SPICE netlists; therefore they can also be used for verification of the transistor level design.

If the transistor level design was verified by simulation the layout can be developed. With the layout level the top down design flow is finished. The layout design is not within the scope of this book. It is possible to extract parasitic effects from layout level simulation which can be used to improve the accuracy of transistor level simulation.

## 2.3 Bottom-up Verification

The amount of information and number of parameters increases during the top-down design process from the system concept to its implementation.

At the beginning of the design, the system is described with some algorithms. After implementation the system may consist of a large number of transistors. Concept verification is needed to check that the implementation meets the requirements of the system.

In the “V” diagram (Figure 2-5) the verification starts from the layout level (bottom) and then proceeds up to the block and system levels.

After layout, simulation parasitic effects can be back-annotated into the circuit netlist. The circuit simulation with the extracted netlist is used to verify the circuit design. The designed circuits can now be combined into functional blocks, which are checked against their specification in a block level simulation. Finally the designed blocks can be connected to the system. System level simulation verifies that the blocks fit into the system environment.

It is recommended to start verification before the design is completed at layout level. After each design step simulation can be used to verify the design or component against the specification.

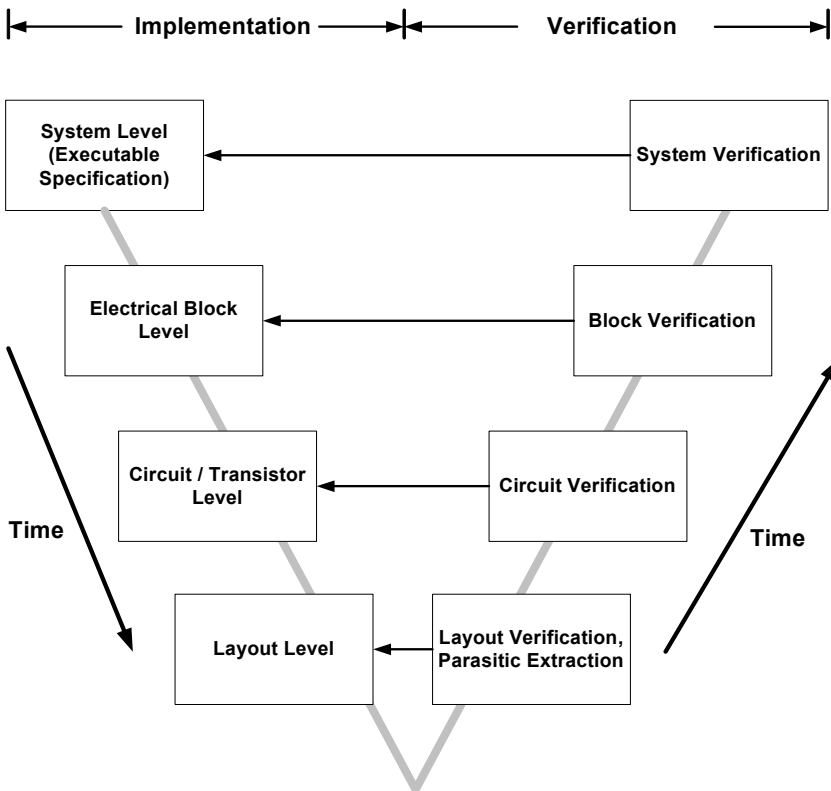


Figure 2-5. Top-down design and bottom-up verification (V diagram)

System level or block level simulation is used to verify large systems or circuits. Often a transistor level model of a system cannot be simulated because its complexity (number of transistors or gates) is much too large. Therefore it is necessary to use behavioral models.

Figure 2-6 shows the application of behavioral models during block level and system level verification. It is assumed that behavioral models were already used during the top-down design. In the verification phase it is now necessary to calibrate these models as follows:

- Parasitic extraction and back annotation into the circuit netlist improves the accuracy of the circuit model (extracted circuit model)
- Simulation with the extracted circuit model is used to gain the circuit characteristic and parameters
- Extracted circuit parameters are used to calibrate the behavioral model of this component
- Calibrated behavioral models are used on block and system levels for verification

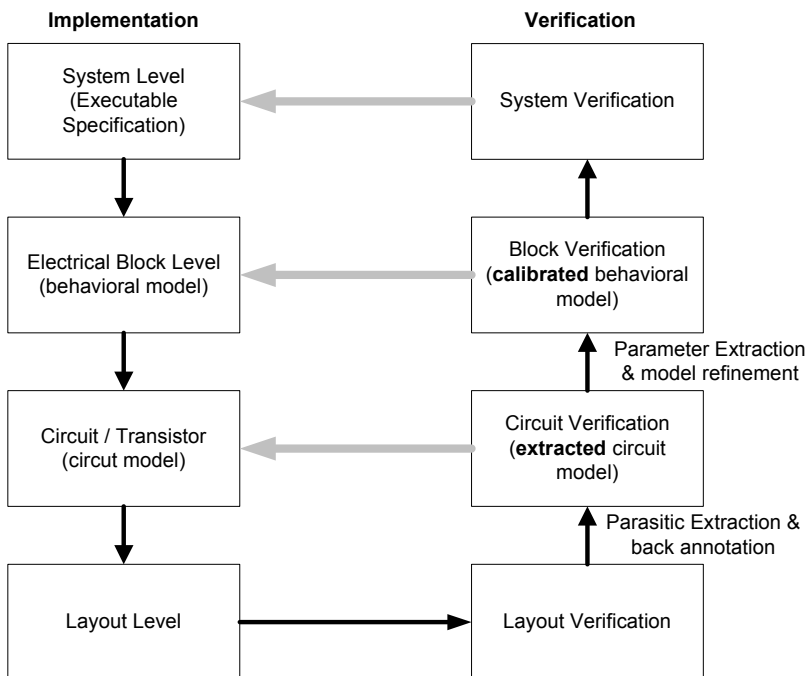


Figure 2-6. Refinement of models during bottom-up verification

The main advantage of using (calibrated) behavioral models is the simulation speedup which enables the simulation of large systems or subsystems.

Different behavioral modeling languages exist. Most of them are specific to a particular simulator. To allow the reuse of models it is suggested to use standardized languages like VHDL-AMS and Verilog-AMS.

A characterization environment can support model calibration. Characterization is the calculation of component or subsystem characteristics and parameters from measured or simulated data. A characterization run contains a set of simulation and postprocessing commands that allow the determination of significant circuit characteristics. The behavior of the circuit description and behavioral model can be compared. If the model is inaccurate, the model parameters or algorithms are modified. Characterization also supports model and circuit documentation. Chapter 11 contains more information about characterization environments.

## Chapter 3

# SIMULATION TOOLS IN SYSTEM DESIGN

### 3.1 Use of Simulation Tools within the Design Flow

The application of simulation tools is very important to improve the efficiency in system and circuit design. Various simulation tools exist on the market to support the design process. This chapter discusses topics that must be taken into account when selecting appropriate simulation tools.

As described in Section 2.2 the top-down design flow starts with the system concept which covers the complete system. The system is then divided into subcomponents down to the circuit and layout level. The choice of simulation tool depends on the design level addressed and the type of design (analog, RF, digital or mixed-signal). Simulators may cover more than one design level (Figure 3-1).

We distinguish between four categories of simulators, which are described in the following sections.

#### **System level simulators**

System level simulators provide efficient simulation algorithms to achieve a high simulation speed. This allows simulation of complete transmission systems containing a transmitter, channel and receiver with analog and digital parts. The simulation accuracy is restricted particularly for analog system parts. However, it allows the verification of system concepts. System modeling is supported by large libraries, which contain models of various system components, for example coders, modulators, and channels. The primary application of these tools is the system level design, also called concept engineering. They may also be partially used in block level design, for example to provide testbenches.

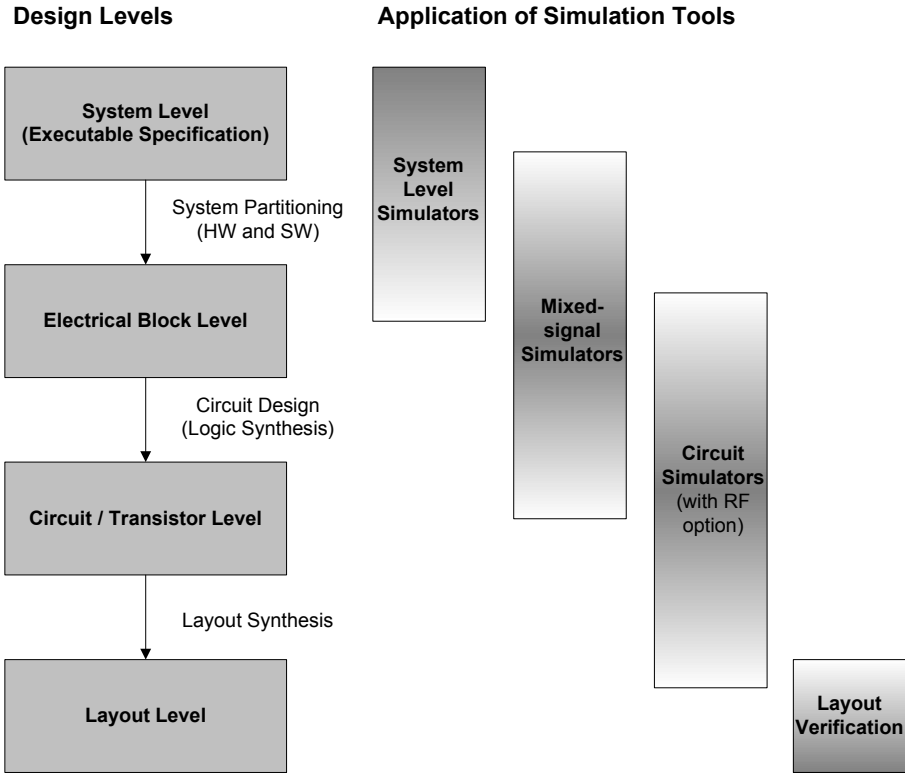


Figure 3-1. Simulation tool coverage in the mixed-signal design flow

### Mixed-signal simulators

The main application of mixed-signal simulators is within the block level design where the partitioning into analog and digital hardware or software is performed. Mixed-signal simulation allows the common verification of analog and digital system parts, as well as the interfaces between them. Behavioral models are widely used at this design stage. The most important mixed-signal modeling languages are VHDL-AMS and Verilog-AMS.

The application of mixed-signal simulators can be extended to the system level if models of the system components exist. However, at present the model libraries of mixed-signal simulators do not achieve the complexity of the system level simulator libraries.

Mixed-signal simulators may also be used in circuit level design. In contrast to specialized RF circuit simulators they do not provide RF specific analyses.

### Circuit level simulators

Most circuit level simulators support the simulation of circuit level descriptions (SPICE netlists) as well as analog behavioral models. Some simulators provide specialized simulation algorithms for the analysis of RF components (circuit envelope, periodic steady state for example). They provide an accurate analysis of components, but the simulation performance is too low to simulate large system parts.

With the ability to use behavioral models, circuit level simulators may also be used in block level design of analog subsystems. In addition layout effects can be included in circuit simulation by extraction of parasitics.

### Layout verification

Layout verification is used to check if the design rules for a desired silicon technology are fulfilled. Layout effects (for example parasitic capacitances, substrate coupling) may be extracted and back annotated for circuit level simulation. The impact of layout and packaging on the desired circuit functionality can be analyzed. Layout verification is not discussed further.

*Table 3-1. Overview of simulation tools*

Simulator	Main design level	Additionally supported levels	Target	Examples
system simulator	system level	block level	complete system	ADS, CoCentric, MATLAB, SPW
mixed-signal simulator	block level	system level, circuit level	subsystems	ADVance MS, SMASH, AMS Designer, Saber
circuit simulator	circuit level	block level, (layout level)	blocks	Eldo, Spectre, Spice, ADS
layout simulator	layout level		components, packages	Assura, Calibre, Hercules

Some simulators and their application are outlined in Table 3-1. In some cases a co-simulation of different tools is used to accelerate the simulation, reuse models, or increase simulation accuracy. This topic is outlined in Chapter 12.

## 3.2 Specific Simulation Algorithms of RF Simulators

The traditional SPICE analyses are essential in analog circuit design. Their application to RF circuits may cause some problems resulting from the behavior of RF systems such as:

- The signals which are transmitted are narrowband signals. This means that a data signal with a relatively low bandwidth is transmitted at a very high carrier frequency. To simulate a sufficient portion of the data signal a large number of carrier waves must be simulated. This may exceed the performance of traditional transient analyses (memory and time consumption).
- RF receivers usually receive weak desired signals while large interference signals are present. This implies that the linearity of the receiver is a very important task for the designer requiring a precise simulation of nonlinearity.
- Improved transistor models are required to represent the behavior of RF transistors.

Specialized RF simulation algorithms are provided to improve the analysis of RF circuits and systems. They are available in RF simulators like ADS and SpectreRF but typically not in VHDL-AMS simulators. An exception is ADMS RF which combines ADVance MS and Eldo RF. The most important simulation algorithms are:

- Periodic Steady State Analysis (PSS)
- Harmonic Balance (HB)
- Transient Envelope Analyses (Envelope)

They provide a good accuracy for RF specific measurements at a sufficient simulation performance. The principle of these analyses is outlined in the following section.

### **Analysis for dynamic systems with weak nonlinearities**

Different simulation algorithms can be used to analyze the frequency response of dynamic and nonlinear systems such as mixers and LNA's. The algorithms are:

- Periodic Steady State (PSS) in Cadence's SpectreRF Simulator
- Harmonic Balance (HB) in Agilent's ADS

The results of these analyses are the frequency spectra of the signals within the system including the wanted and unwanted harmonics (arising from nonlinearity).

The analysis is used to compute the steady state response of a nonlinear circuit, which is the response after the start-up transient has died down. The stimulus of the circuit is a limited number of sinusoidal signals. In the steady state, the system response is periodic according to the period length of the

fundamental frequency. All input frequencies of the system must be an integer multiple of the fundamental frequency. The methods of computing the steady state solution are different in PSS and HB.

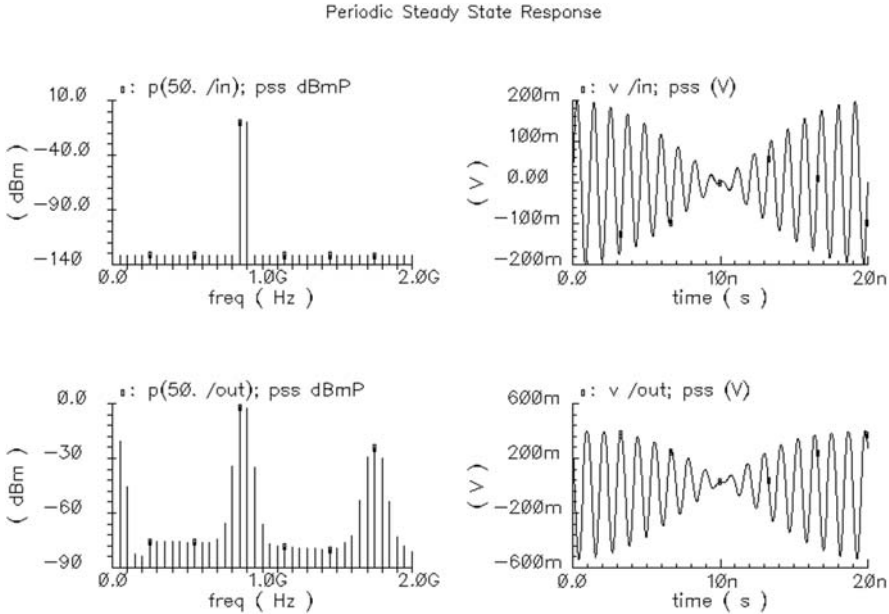


Figure 3-2. Results of a PSS analysis of an LNA

Figure 3-2 shows the results of a PSS analysis in frequency (left hand graphs) and time domain (right hand graphs). The input signal was two-tone with 850 and 900 MHz, each with a -10 dBm magnitude (upper graphs). Each input frequency must be an integer multiple of the fundamental frequency. Thus a fundamental frequency of 50 MHz is used in the example. This is equivalent to a period of 20 ns. To visualize frequencies up to 2 GHz, 40 harmonics of the fundamental frequency were computed. The time domain output of the LNA (bottom right hand graph) shows that the LNA is operated in the nonlinear area. The 3rd order harmonics at 800 MHz and 950 MHz are visible in the frequency domain (upper left hand graph). Other analyses are based on the steady state operating point, for example:

- periodic AC analysis
- periodic noise analysis
- periodic XF (periodic transfer function)
- periodic SP (periodic S-parameters)

The PSS analyses and the subsequent analyses are very important to determine the characteristics of RF systems and building blocks.

### Transient envelope analyses

The envelope analyses address the narrow-band problem of wireless communication systems: signals with a relatively small bandwidth are transmitted at very high carrier frequencies. Transient envelope analyses are known as:

- Circuit Envelope Analysis (ADS from Agilent)
- Envelope Following Analysis (SpectreRF from Cadence)

The transient envelope analysis computes the envelope of a modulated carrier signal. This is demonstrated with a sine wave of 1 MHz, which is amplitude modulated on a carrier frequency of 900 MHz (modulation index 0.5). The simulation interval is 2  $\mu$ s (two periods of the modulation signal).

Figure 3-3 shows the AM modulated carrier resulting from a transient analysis. To represent the modulated signal a large number of carrier periods must be computed, which is visualized in the detail interval (1...1.02  $\mu$ s). This implies that the transient analysis is not efficient enough to evaluate a sufficient part of the modulation signal. The transient envelope analysis can speed-up the simulation of the modulation signal.

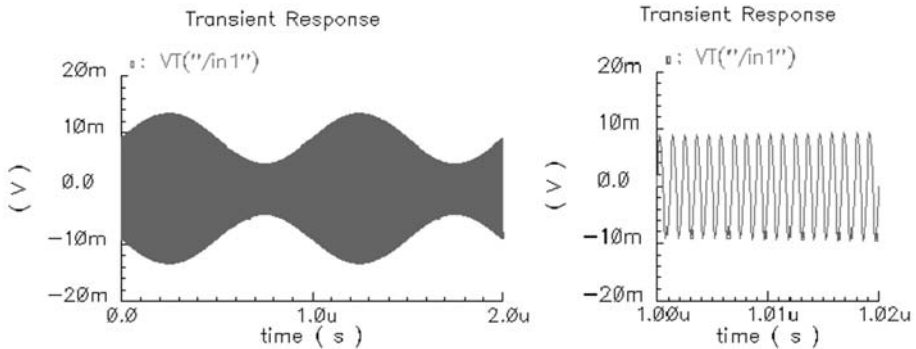


Figure 3-3. Results of traditional transient analyses (complete wave and detail)

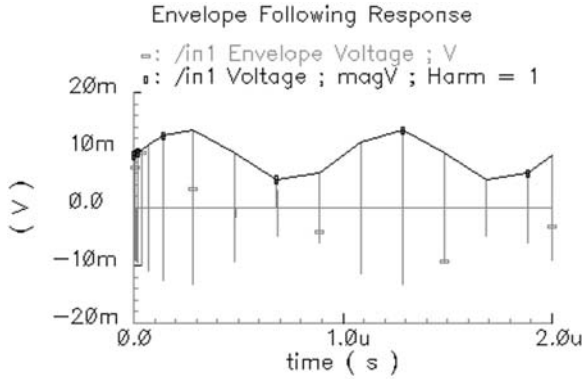


Figure 3-4. Result of the envelope following analysis (SpectreRF)

The envelope analysis was six times faster than the transient analysis of a small example LNA. The lower portion of the graph in Figure 3-4 shows the time domain signal of the modulated carrier. It can be seen, that the carrier signal is only partially computed. The black curve shows the envelope of the carrier which represents the modulating signal. There are too few sampling points to achieve a clear sine wave. The envelope analyses may be hardly applicable for multi-carrier or wideband modulation techniques.

### 3.3 Criteria of the Simulator Selection

A great number of simulation tools are on the market. This section presents some criteria which must be taken into consideration to identify the best simulation tool for a design task. The decision depends on the application, design flow, user interface, costs, and support.

#### Application related criteria

- In which design level(s) should the simulator be used?
- Which designs shall be mostly simulated (analog, mixed-signal)?
- Are special analyses needed (for example for RF)?
- Which model libraries are provided to speed-up the modeling of systems and testbenches?
- Is it possible to reuse models of former designs?
- Which simulation speed can be obtained?
- Is the size of the designs limited?

**Design flow related criteria**

- Are there interfaces for standardized modeling languages?
- Are there interfaces to other tools in the existing design flow (model import/export, simulator coupling)?
- Are there interfaces for tool customization and scripting?
- Is version control supported?
- Which computing platforms are supported (Windows, Unix, Linux, others)?

**User interface related criteria**

- Is a graphical user interface available?
- Schematic or netlist entry or both?
- Quality of documentation? (User guides, examples, reference manuals, tutorials, ...)

**Cost related criteria**

- Costs of licenses? (buying, leasing, public domain)
- Costs of support and version update?
- Time that is needed for user training?
- Costs of user training?
- Time/costs for software installation and maintenance?

**Support Related Criteria**

- Software support available?
- Web based support databases?
- Design service (special support on user applications)?

The criteria mentioned above shows that the selection of a simulation tool is very difficult. The integration of a new simulation tool often depends on the existing design flow. Some major vendors of EDA tools provide design frameworks where different tools have been integrated with a common user interface.

In the future, interfaces for standardized modeling languages, like VHDL-AMS, will simplify the exchange of models between simulators.